

Advanced Coding Using JavaScript or Python Learning Objectives

Learning Objectives	Learning Outcome	CSTA K-12 CS Standards
P3.1) Identify a scenario where you could reuse existing blocks or code to accomplish a task	Learners will understand that code is reusable	2-AP-16 Incorporate existing code, media, and libraries into original programs, and give attribution. 2-AP-14 Create procedures with parameters to organize code and make it easier to reuse.
P3.2) Given some code identify iteration and section blocks in MakeCode and JavaScript code and explain the use and results of each block	Learners will be able to identify iteration and selection in both MakeCode blocks and JavaScript	2-AP-12 Design and iteratively develop programs that combine control structures, including nested loops and compound conditionals.
P3.3) Following a set of written instructions determine the outcome	Learners will be able to demonstrate the ability to determine the output of a program given only the program and its input without running the program	2-AP-10 Use flowcharts and/or pseudocode to address complex problems as algorithms.
P3.4) Identify the blocks of JavaScript which relate to the equivalent MakeCode blocks	Learners will be able to identify the equivalent JavaScript code of MakeCode blocks	2-AP-12 Design and iteratively develop programs that combine control structures, including nested loops and compound conditionals.

CODING IN MINECRAFT



Learning Objectives	Learning Outcome	CSTA K-12 CS Standards
P3.5) Given a problem, design and develop a solution using MakeCode	Learners will be able to design and develop a programmatic solution for a given problem	2-AP-13 Decompose problems and subproblems into parts to facilitate the design, implementation and review of programs.
P3.6) Given a series of instructions with syntax errors identify the syntax errors and provide solutions	Learners will be able to identify syntax errors in existing code and provide explanations of the syntax error and suitable resolution(s)	2-AP-19 Document programs in order to make them easier to follow, test, and debug. 2-AP-17 Systematically test and refine programs using a range of test cases.
P3.7) Incorporate existing code and use modularity	Demonstrate the ability to incorporate existing code and comment code appropriately to give attribution Demonstrate the ability to decompose problems into subproblems and create and use procedures	2-AP-13 Decompose problems and subproblems into parts to facilitate the design, implementation, and review of programs. 2-AP-16 Incorporate existing code, media, and libraries into original programs, and give attribution.